International Journal of Trend in Scientific Research and Development (IJTSRD)
Volume 5 Issue 1, November-December 2020 Available Online: www.ijtsrd.com e-ISSN: 2456 – 6470

CipherKey Algorithm

Sweety Gone[1], Kuldeep B. Vayadande[2]

[1]Master Student, Department of MCA, [2]Assistant Professor,
[1,2]Jain Deemed-to-be University, Bengaluru, Karnataka, India

ABSTRACT

Converting a plain text into non-readable format to maintain a confidentiality & integrity of data is called Encoding. And the technique used to decode that into readable format, is called Decryption. To encrypt & decrypt, algorithms we have developed. This entire theory, The whole technology is called Cryptography. Many algorithms were developed, many are Decoded, and many of the algorithms are still running nowadays also. So, here I came up with the new algorithm, with new technique, with new idea in algorithm.

Keywords: *Information Security, Integrity, Encryption, Decryption, Symmetric Algorithm, Cipher*

How to cite this paper: Sweety Gone | Kuldeep B. Vayadande "CipherKey Algorithm" Published in International Journal of Trend in Scientific Research and Development (ijtsrd), ISSN: 2456-6470, Volume-5 | Issue-1, December 2020, pp.388-392, URL: www.ijtsrd.com/papers/ijtsrd37924.pdf

IJTSRD37924

1. INTRODUCTION

In Information Security, Encipher & Decode is used to maintain data purity & Private. Cryptography is all about encryption & decryption. And cryptography comes under cryptology. In Cryptography, algorithms are there, through which security takes place. Algorithms are of 2 types: 1) Symmetric Algorithm & 2) Asymmetric algorithm. In symmetric algorithm, public key encryption concept is used & in Asymmetric algorithm, public key - private key algorithm concept is used. According to security researchers, symmetric key algorithm process fast as compare to public key (asymmetric key) algorithm because, symmetric algorithm can't use lengthy mathematical logics & concepts. So, here I came with a new technique of key algorithm to encrypt a plain key to make secure communication. It's just converting a key to encrypted form. It's a technique which converts 16 characters to 192 characters. Yes, it is like encoding 128 bits to 1536 bits of encryption. And I am using here 8 x 8 of matrix. Means 64 characters is there; 26 Lowercase (small) alphabets, 26 Uppercase (CAPITAL) alphabets, 0-9 (10 numerical digits), 2 special characters. And I'm converting by doing 12 rounds of matrix & their combinations.

This is a symmetric key algorithm, where the key will be in encrypted form, and only the receiver who has this application or the one who knows the algorithm flow can decrypt it. This algorithm is a kind of Playfair algorithm, Caesars Cipher and ROT13 algorithm but not same like that. I referred similar kind of concepts of Playfair Algorithm, Caesars Cipher and ROT13 algorithm.

2. PROPOSED SYSTEM

Proposed systems from where I got to know that they can be an unsafe and some weakness which I've noticed. That proposed algorithms are as follows:

A. Caesar Cipher
B. Playfair Algorithm
C. ROT13 Algorithm

Now, let's discuss about these algorithms & their weakness which I noticed:

Caesar Cipher

Caesar cipher algorithm is an algorithm which performs shifting of alphabets with some certain number of positions from down or up alphabets. It's a kind of substitution cipher. Example:

Suppose, if shifting position number is 3, then A would be replaced by D. And the position is permanent for the entire key string.

Text: ABC
Shift: 3
Cipher: DEF

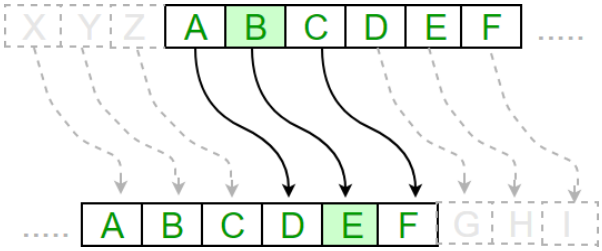

Fig 1: Caesar Cipher

Source: https://www.geeksforgeeks.org/caesar-cipher-in-cryptography

As we can see, 26 alphabets are there. So, there are 26 possibilities is there to crack the cipher key. This algorithm is very easy to crack . Now what is cipher key? "The string result which we get after encoding is called Cipher Key."

Play fair Algorithm

Play fair algorithm is also a exchange cipher. The concept of play fair algorithm is, it swaps two alphabets position with each other in row-wise & column-wise and if both are not possible then it'll form a rectangle with 2 letters & swap the letters with horizontal corner values. Matrix is of 5x5, means 25 alphabets is there but actual alphabets are of 26 characters. The J will be uniting with I. And the key string will be firstly distributed in matrix after that remaining alphabets will be distributed.

E.g.: Suppose key is **monarchy,** then matrix will be like:

M	O	N	A	R
C	H	Y	B	D
E	F	G	I	K
L	P	Q	S	T
U	V	W	X	Z

Fig 2: Playfair Algorithm

Source: https://www.geeksforgeeks.org/playfair-cipher-with-examples/

Plain text: instruments
Split in: „in','st','ru','me','nt','sz'

We have add 'z' extra because we want in even alphabets.

Rules for Encryption:

1. If both the characters are in same column, then next alphabet to the present alphabet will be swapped vertically, as follows:

M	O	N	A	R
C	H	Y	B	D
E	F	G	I	K
L	P	Q	S	T
U	V	W	X	Z

Fig 3: 1st Rule of Playfair Algorithm

Source: https://www.geeksforgeeks.org/playfair-cipher-with-examples/

2. If both alphabets are in same row, then next alphabet to the present alphabet will be swapped horizontally, as follows:

M	O	N	A	R
C	H	Y	B	D
E	F	G	I	K
L	P	Q	S	T
U	V	W	X	Z

Fig 4: 2nd Rule of Playfair Algorithm

Source: https://www.geeksforgeeks.org/playfair-cipher-with-examples/

3. If both will be not there, then rectangle will be created, as follows:

M	O	N	A	R
C	H	Y	B	D
E	F	G	I	K
L	P	Q	S	T
U	V	W	X	Z

Fig 5: 3rd Rule of Playfair Algorithm

Source: https://www.geeksforgeeks.org/playfair-cipher-with-examples/

ROT13 Algorithm

ROT13 means rotate 13 positions. ROT13 algorithm is same like Caesar cipher algorithm but shifting position is always 13, it"ll never change.

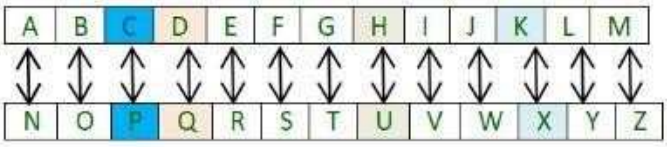

Fig 6: Overview of ROT13

Source: https://www.geeksforgeeks.org/playfair-cipher-with-examples/

If key is **Sweety,**

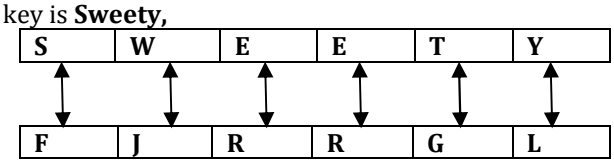

Fig 7: Example of ROT13

You're thinking what's the difference between Caesaer Cipher & ROT13? Difference is execution of algorithm. So, cracking this algorithm is possible, if analyst or hacker try all the 26 possibilities to crack. So, getting these all possibilities which can crack these algorithms, I've joined concepts of Caesar Cipher, Playfair Algorithm & ROT13. After that I added some rounds of AES Algorithm. Now you are thinking what AES algorithm is & what are rounds? AES algorithm is the asymmetric key algorithm where public key & private key system is used. An AES algorithm generates encryption by performing same process repeatedly. It's like looping of process till some limit. So, here also I've executed this technique. Because of this, I given name to this algorithm, which is called as CipherKey Algorithm? It creates 1536 bits of key which is very difficult to crack.

3. FLOWCHART

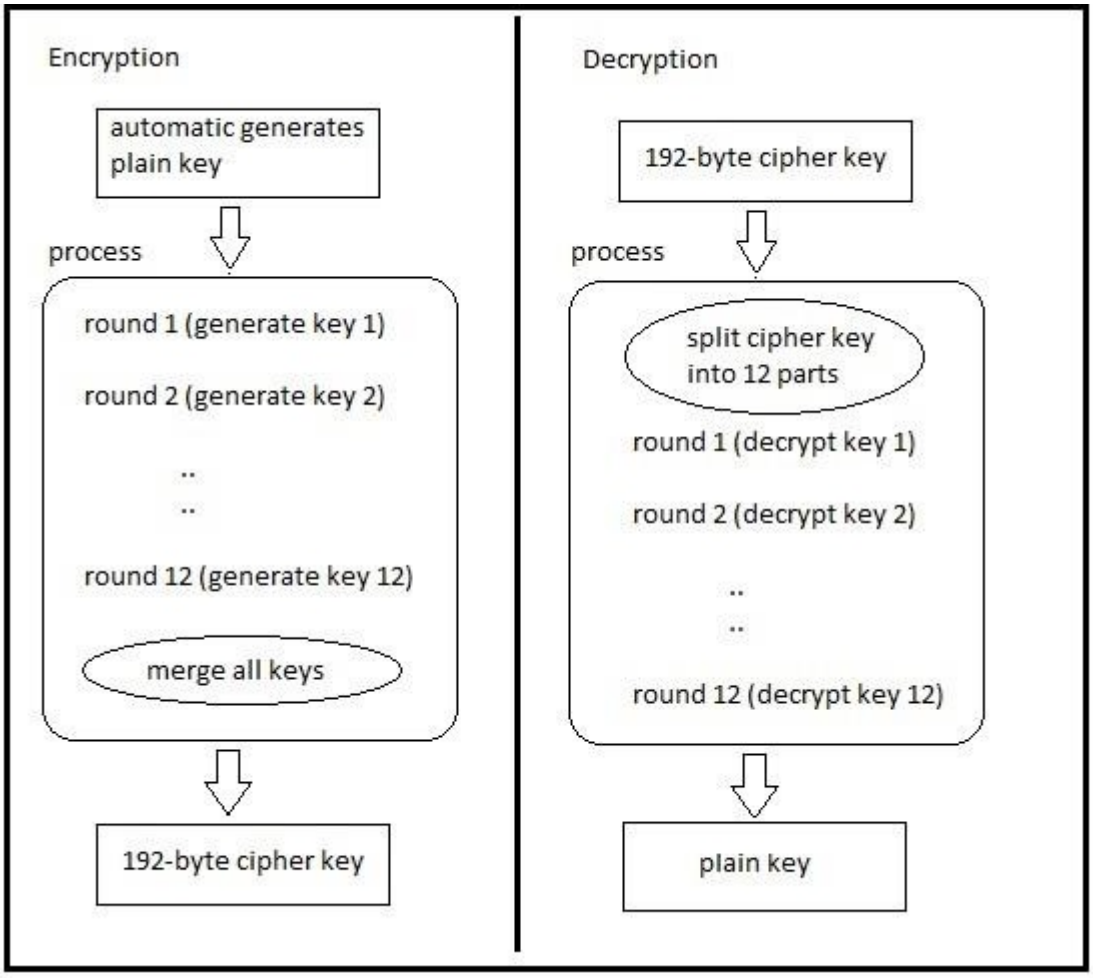

Fig 8: Flowchart of CipherKey Algorithm

4. PROBLEM WITH PROPOSED SYSTEM

I've created this new algorithm which is based oon playfair algorithm. The problem with playfair algorithm is that, key will easily know by any cracker, because while encrypting that key will be put first row, then remaining blanks will be filled with remaining characters. Like if key is "MONARCHY". Then it will be stored in 5*5 matrix is like:

M	O	N	A	R
C	H	Y	B	D
E	F	G	I	K
L	P	Q	S	T
U	V	W	X	Z

J is not there in this table but it will be with I like this I/J

But in my matrix, key stored at different places like 1st, middle and last, column-wise & rows-wise. It's not the same like Playfair, kind of, like changing positions. Cracking playfair cipher possibilities are 625 (25*25).

5. PROCEDURE FOR ALGORITHM (ENCRYPTION)

1. Take a string of set of all Uppercase(A-Z) Alphabets, Lowercase(a-z) alphabets, Numerical (0-9) digits, Special characters(@,#) these are the string which I've taken.
e.g.:Str="ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789@#"
2. Convert string & store it into 1D-Array(1 Dimensional array)
3. Convert 1D-Array into 2D-array(2 Dimensional array)

A	B	C	D	E	F	G	H
I	J	K	L	M	N	O	P
Q	R	S	T	U	V	W	X
Y	Z	a	b	c	d	e	f
g	h	i	j	k	l	m	n
o	p	q	r	s	t	u	v
w	x	y	z	0	1	2	3
4	5	6	7	8	9	@	#

Table 2: Convertion of 1-Dimensional array to 2-Dimensional array

4. Take characters from particular positions(0,3,4,7) row-wise & column-wise, and store it into 2D-Array
P=array of position(0,3,4,7)
A=1D-array
K=key

```
loop I of A row-wise loop X of P
    if P == I
        loop J of A [ I ] column-wise loop Y of P
            if J == Y
                K = K + value at position[I][J]
```

At the end we'll get 16 digit key. And it will be stored in 2D-Array.

5. Swap 1st row with 4th Row &
   5th Row with 8th Row
   repeat step 4

6. Swap 1st column with 4th column &
   5th column with 8th column
   repeat step 4
7. Swap 1st row values with last row values,
   2nd row with (last – 1) row,
   3rd row with (last – 2) row,
   4th row with (last – 3) row,
   and repeat step 4

8. Swap 1st column with last column,
   2nd column with (last – 1) column,
   3rd column with (last - 2) column,
   4th column with (last – 3) column,
   and repeat step 4

9. Swap 1st row values with 5th row values,
   2nd row with 6th row,
   3rd row with 7th row,
   4th row with 8th row,
   and repeat step 4

10. Swap 1st column with 5th column,
    2nd column with 6th column,
    3rd column with 7th column,
    4th column with 8th column,
    and repeat step 4

11. Now, in selected block, increase ASCII values with 1

A	B	C	D	E	F	G	H
I	J	K	L	M	N	O	P
Q	R	S	T	U	V	W	X
Y	Z	a	b	c	d	e	f
g	h	i	j	k	l	m	n
o	p	q	r	s	t	u	v
w	x	y	z	0	1	2	3
4	5	6	7	8	9	@	#

Table 3: Select block from array

And repeat step 4

12. Now, in selected block, increase ASCII values with 2

A	B	C	D	E	F	G	H
I	J	K	L	M	N	O	P
Q	R	S	T	U	V	W	X
Y	Z	a	b	c	d	e	f
g	h	i	j	k	l	m	n
o	p	q	r	s	t	u	v
w	x	y	z	0	1	2	3
4	5	6	7	8	9	@	#
w	x	y	z	0	1	2	3
4	5	6	7	8	9	@	#

Table 4: Select block from array

And repeat step 4

13. Now, in selected block, increase ASCII values with 1

A	B	C	D	E	F	G	H
I	J	K	L	M	N	O	P
Q	R	S	T	U	V	W	X
Y	Z	a	b	c	d	e	f
G	h	i	j	k	l	m	n
O	p	q	r	s	t	u	v
W	x	y	z	0	1	2	3
4	5	6	7	8	9	@	#

Table 5: Iincreasing Array after ASCII values

And Repeat step 4

14. Now, in selected block, increase ASCII values with 2

A	**B**	**C**	**D**	**E**	**F**	**G**	**H**
I	**J**	**K**	**L**	**M**	**N**	**O**	**P**
Q	**R**	**S**	**T**	**U**	**V**	**W**	**X**
Y	**Z**	**a**	**b**	**c**	**d**	**e**	**f**
g	**h**	**i**	**j**	**k**	**l**	**m**	**n**
o	**p**	**q**	**r**	**s**	**t**	**u**	**v**
w	**x**	**y**	**z**	**0**	**1**	**2**	**3**
4	**5**	**6**	**7**	**8**	**9**	**@**	**#**

Table 6: Increasing Array after ASCII values

And Repeat step 4

15. Again repeat step 14, increase ASCII value with +2

16. After this all, we'll get 2D-array, full of 12 keys from 12 rounds, as follows:

A. ADEHY12569aduxy#
B. Y125ADEHuxy#69ad
C. 1Y52DAHExu#y96da
D. 96daxu#yDAHE1Y52
E. ad69y#uxEHAD25Y1
F. EHAD25Y1ad69y#ux
G. ADEHY12569aduxy#
H. BEEHZ22569aduxy#
I. BEGJZ24769aduxy#
J. BEGJZ2477:advyy#
K. BEGJZ2477:cfvy{%
L. BEGJZ2477:cfvy}"

Now, we've to joint this all keys. So, we can ge encoded keys. But, before uniting of all keys, we have to make it complex. Because last key is needed to start decoding & 1st one to get plain key.

Now, replace 1st position with 6th position & 7th position with 12th position. Will be getting following results.

1. EHAD25Y1ad69y#ux
2. Y125ADEHuxy#69ad
3. 1Y52DAHExu#y96da
4. 96daxu#yDAHE1Y52
5. ad69y#uxEHAD25Y1
6. ADEHY12569aduxy#
7. BEGJZ2477:cfvy}"
8. BEEHZ22569aduxy#
9. BEGJZ24769aduxy#
10. BEGJZ2477:advyy#
11. BEGJZ2477:cfvy{%
12. ADEHY12569aduxy#

Now, unite all keys, 192-byte encrypted key will be generated, which look like:

EHAD25Y1ad69y#uxY125ADEHuxy#69ad1Y52DAHExu#y96da
96daxu#yDAHE1Y52ad69y#uxEHAD25Y1ADE
HY12569aduxy#BEGJZ2477:cfvy}'BEEHZ22569aduxy#BEGJZ
24769aduxy#BEGJZ2477:advyy#BEGJZ2477:cfv
y{%ADEHY12569aduxy#

6. PROCEDURE FOR ALGORITHM (DECRYPTION)

192-byte key needed first,
EHAD25Y1ad69y#uxY125ADEHuxy#69ad1Y52DAHExu#y96da
96daxu#yDAHE1Y52ad69y#uxEHAD25Y1ADE
HY12569aduxy#BEGJZ2477:cfvy}'BEEHZ22569aduxy#BEGJZ
24769aduxy#BEGJZ2477:advyy#BEGJZ2477:cfv
y{%ADEHY12569aduxy#

Convert this one string into 2D-array, as follows:

1. EHAD25Y1ad69y#ux
2. Y125ADEHuxy#69ad
3. 1Y52DAHExu#y96da
4. 96daxu#yDAHE1Y52
5. ad69y#uxEHAD25Y1
6. ADEHY12569aduxy#
7. BEGJZ2477:cfvy}'
8. BEEHZ22569aduxy#
9. BEGJZ24769aduxy#
10. BEGJZ2477:advyy#
11. BEGJZ2477:cfvy{%
12. ADEHY12569aduxy#

From 192-byte key, last 16-digit key is needed to start decryption.

But before it, we've to put values at their own position. Swap 1st values with 6th value & 7th value with 12th value. It will look like,

1. ADEHY12569aduxy#
2. Y125ADEHuxy#69ad
3. 1Y52DAHExu#y96da
4. 96daxu#yDAHE1Y52
5. ad69y#uxEHAD25Y1
6. EHAD25Y1ad69y#ux
7. ADEHY12569aduxy#
8. BEEHZ22569aduxy#
9. BEGJZ24769aduxy#
10. BEGJZ2477:advyy#
11. BEGJZ2477:cfvy{%
12. BEGJZ2477:ehvy}'

Now, pick 12th position value, and proceed to decryption.

1. Convert key string into 1D-array.
2. Make a loop of 4*4

B	E	G	J
Z	2	4	7
7	:	e	H
V	y	}	'

Table 7: Array of key

3. Remove table ASCII values with –
   2 Key: BEGJZ2477:cfvy{%
4. Again, remove table ASCII values with –
   2 Key: BEGJZ2477:advyy#
5. Remove table ASCII values with –
   1 Key: BEGJZ24769aduxy#
6. Remove table ASCII values with –
   2 Key: BEEHZ22569aduxy#
7. Remove table ASCII values with –
   1 ADEHY12569aduxy#
8. Swap left 2 columns with right 2 columns, like swapping 1st position table-column with 3rd position table- column & 2nd position table-column with 4th position table-column Key: EHAD25Y1ad69y#ux
9. Swap top 2 rows with bottom 2 rows, like swapping 1st position table-row with 3rd position table-row & 2nd position table-row with 4th position table-row Key: ad69y#uxEHAD25Y1
10. Swap 1st column with 4th column, 2nd column with 3rd column Key: 96daxu#yDAHE1Y52
11. Swap 1st row with 4th row, 2nd row with 3rd row Key: 1Y52DAHExu#y96da
12. Swap 1st column with 2nd column, 3rd column with 4th column Key: Y125ADEHuxy#69ad
13. Swap 1st row with 2nd row, 3rd row with 4th row Key: ADEHY12569aduxy#

Now finally, we got our key 16-digit key which is ADEHY12569aduxy#

7. CONCLUSIONS

CipherKey creates algorithm in the strongest & unbreakable. Plain key will be automatic generated every time. So, for decoder it's difficult to crack. By using this algorithm, integrity & confidentiality should be maintained. If bits are too extended then it will be difficult to decode.

8. REFERENCES

[1] Network Security & Cryptography by Atul Kahate

[2] Websites:
   A. www.geeksforgeeks.com
   B. www.javatpoint.com
   C. www.tutorialspoint.com
   D. https://people.eecs.berkeley.edu/~bh/pdf/v1ch12.pdf
   E. https://en.wikipedia.org/wiki/Category:Cryptographic_algorithm